

REMARKS

This is intended as a full and complete response to the Office Action dated July 29, 2004, having a shortened statutory period for response set to expire on October 29, 2004. Please reconsider the claims pending in the application for reasons discussed below.

In the specification, the abstract, Paragraphs 5 and 9 have been amended to correct minor editorial problems as suggested by the Examiner. In addition, Paragraphs 25, 34, 43, 46, 47, 49, 51, and 43 have also been amended to correct minor editorial problems, and to include a "®" trademark attribution where appropriate. Applicants' assert that none of the amendments submitted introduce any new matter into the specification.

Claims 1-33 are pending in the application. Claims 1-4, 7-15, 17-36 remain pending following entry of this response. Claims 1, 14, 19, 25, 28, 30 and 32 have been amended. Claims 5-6, 16 and 29 have been cancelled. New claims 34-36 have been added to recite aspects of the invention.

Claim 19 stands rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. The Examiner states that Claim 19 contains the trademark/trade name Netscape Navigator® and Microsoft Internet Explorer®. Applicants have amended claim 19 to address the rejection.

Claims 1-3, 5-7, 9-12, 14-17, 19, 21-26 and 29-33 stand rejected under 35 U.S.C. § 102(e) as being anticipated by *Friskel*, et al. U.S. Patent 6,683,629, (hereinafter "*Friskel*") under 35 U.S.C. § 102(e) Applicants respectfully traverse the rejection.

Friskel discloses a method for embedding a child window in a parent window. Generally, a window defined by an add-on program is displayed to a user of an application program. The add-on program detects the display to the user of an application window created by the application program. In response to this detection, the add-on program displays the add-on window to the user. (*Friskel*, Abstract) Such

IN THE DRAWINGS:

The attached sheet of drawings includes changes to Figure 2 and Figure 5.

Attachment: Replacement Sheet

windows are commonly known as a “pop-up” window that typically contains (oftentimes unwanted) advertising messages. *Friskel*, Column 4 lines 18-26 (4:18-26).

The add-on program disclosed by *Friskel* is configured to detect certain operating system events (*Friskel*, 4:29-38), or to query the system to determine whether certain objects do, or do not exist (*Friskel*, 5:39-50). “In response to this detection, add-on program will display add-on window on [a] display unit.” *Friskel*, 4:38-43 (reference numbers omitted). In various embodiments described in *Friskel*, the add-on program is used to monitor the activity of a user (by monitoring a target application) and to present the user with additional content in the child window.

Once displayed, the child window operates independently of the add-on program and is a child of the parent application being monitored by the add-on program. That is, the add-program is a software application in which advertisements are displayed using the program being monitored. The add-on program includes additional code that displays the ads in pop-up windows or through a bar that appears on a user’s computer screen. Advertisements are displayed in response to the detection of certain events. Rather than control a particular browsing session, *Friskel* discloses an add-on program that monitors a user’s activity. “The content and other functions presented for viewing and action by the user often will be complementary to those provided by a parent window. However, in other cases it may be desirable that the content and services presented by the add-on window be a competitive substitute to that originally generated and displayed by application program. As an example, a service provider that distributes add-on program may desire to compete with products and or services offered to the user by windows displayed by application program.” *Friskel*, 4:18-27 (reference numbers omitted).

The add-on disclosed by *Friskel* program typically comprises several files, including a main executable file, for executing various functional aspects of the program and defining the content or the location of the content, whether local or remote, to be displayed in add-on window. *Friskel*, 3:61-65. Thus, the teachings of *Friskel* require an additional computer program executing on the users’ system.

Referring to claims 1, 14 and 25, in contrast to the teachings of *Friskel*, Independent claims 1, 14, and 25 each recite the limitation of opening a controlling browser window configured to control aspects of a controlled browser window; and opening the controlled browser window comprising a display area for rendering viewable content received from network locations. The control comes from the operations of the operations of the controlling browser window, and not from an independent program. Regarding claims 1, 14, and 25, the Examiner specifically asserts that:

Friskel et al. teach a method, computer readable medium and computer comprising a memory containing at least a browser programming (column 3, lines 4-34 and Figure 1), a processor (column 3, lines 4-19 and Figure 1) which when executing the browser programming is configured to open a controlling browser window configured to control aspects of a controlled browser window (a parent browser window comprising toolbar buttons used to control the information displayed in application browser windows and add-on browser windows) and open the controlled browser window comprising a display area for rendering viewable content received from network locations (displaying information received from the Internet in the application browser window and add-on browser windows residing in the parent browser window) (column 4, lines 3-23 and 62-67, and column 9, lines 56-67).

Friskel, however, teaches that the add-on program may open child window i.e., a pop-up window. And that the pop-up window may include “pull-down or other types of menus and/or present icons that the user can use to start or initiate other services or functions that may be associated with application 112 or be of other interest to the user.” *Friskel*, 4:9–14. The controls (e.g., the browser chrome) of the parent window remain unaltered, as do the responses that occur when a user interacts with the browser controls. The user remains free to navigate to any desired network location, or interact with any of the controls of the parent window (or child window, if opened with any such controls), and therefore, the add-on program in no way controls the operations of the parent window.

The Examiner further asserts that Figure 9 shows a parent browser window 900 is displayed and comprises application window 912, add-on child browser windows 902, 904 and 906 and toolbar buttons such as back, forward, etc. to control the information

displayed. Applicants' do not claim the general operation of browser windows, like the Multiple Document Interface illustrated in Figure 9. Rather, Applicants' claim a method to control the actions of a user during a particular browsing session using a first, hidden, browser window that controls the actions of a second, controlled, browser window.

Independent claims 1, 14, and 25 each recite a limitation that the controlling browser window comprises a hidden window. The examiner asserts that "*Friskel* teaches opening the controlling browser window comprises opening a hidden window and wherein opening the controlled browser window comprises opening a viewable window (as can be seen from Figure 9, since the child windows are embedded within the parent window, it is the contents of the add-on child windows in the parent window and not the content of the parent window that are viewable)." Respectfully, this mischaracterizes *Friskel*, Figure 9. The description that accompanies this figure provides:

FIG. 9 illustrates an example in a specific embodiment of a parent window with embedded child windows according to the present invention. Specifically, parent window 900 is generated by an application program. Add-on windows 902, 904, and 906 are initially displayed in a docked configuration as shown and are generated by an add-on program as generally described above, for example as child windows embedded in parent window 900. Window 910 is an additional child window generated by the add-on program, which may be opened at the same time as windows 902, 904, and 906. **Window 912 is an application window that is generated by the application program.** Bottom portion 914 of each add-on window 902, 904, and 906 rolls up like a rolling window shade (this is activated when a user, for example, clicks on arrow icon 916 using a mouse) into header bar 918 for each window so that only header bar 918 appears in parent window 900 when in such a rolled-up state.

Friskel, 9:56 – 10:6 (emphasis added). The window 912, includes all the content of the parent window, plainly visible in Figure 9. At most, add-on program displays window banner 910 containing an advertisement that obscures a portion of the parent window. Further, the description specifies that add-on windows 902, 904, and 906 are "initially displayed in a docked configuration," that could easily be undocked by a user, or changed to a "rolled up state" *Friskel*. The add-on program disclosed by *Friskel* only spawns additional windows from the parent window, both of which are plainly visible. Thus, *Friskel* fails to teach the limitation of wherein the controlling browser window is a

hidden window and wherein the controlled browser window is a viewable window. Accordingly, Applicants respectfully assert that the claims 1, 14, 25, and the claims depending therefrom, are allowable, and therefore, request the Examiner to withdraw the rejection

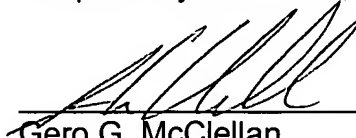
Claims 4, 20 and 28 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over *Friskel* et al. U.S. Patent 6,683,629, as applied to claims 1, 14 and 25 above, and further in view of *Medoff* U.S. Publication 2003/00885 17. Applicants submit that *Friskel*, as applied to claims 1, 14 and 25, is believed to have been overcome for the reasons given above. Accordingly, the combination of *Friskel* and *Medoff* is believed to be obviated. Therefore, the claims are believed to be allowable and allowance of the same is respectfully requested.

Claims 8, 13, 18 and 27 stand rejected under 35 U.S.C. 103(a) as being unpatentable over *Friskel*, as applied to claims 1, 14 and 25 above, and further in view of *Hodgkinson*, U.S. Publication 2002/00 16802. Applicants submit that *Friskel*, as applied to claims 1, 14 and 25, is believed to have been overcome for the reasons given above. Accordingly, the combination of *Friskel* and *Hodgkinson* is believed to be obviated. Therefore, the claims are believed to be allowable and allowance of the same is respectfully requested.

The secondary references made of record are noted. However, it is believed that the secondary references are no more pertinent to the Applicants' disclosure than the primary references cited in the office action. Therefore, Applicants believe that a detailed discussion of the secondary references is not necessary for a full and complete response to this office action.

Having addressed all issues set out in the office action, Applicants respectfully submit that the claims are in condition for allowance and respectfully request that the claims be allowed.

Respectfully submitted,



Gero G. McClellan

Registration No. 44,227

MOSER, PATTERSON & SHERIDAN, L.L.P.

3040 Post Oak Blvd. Suite 1500

Houston, TX 77056

Telephone: (713) 623-4844

Facsimile: (713) 623-4846

Attorney for Applicants